CHAPTER 5

CONSTRAINT SATISFACTION PROBLEM

5.1 INTRODUCTION

Constraint programming falls in the category of declarative programming styles, where essentially a program describes what must be computed, as opposed to imperative programming, where a program describes how the output must be computed. A constraint “program” is called a constraint satisfaction problem (CSP).

In artificial intelligence and operations research, constraint satisfaction is the process of finding a solution to a set of constraints, in which the imposed conditions that the variables must satisfy. A solution is, therefore, a vector of variables that satisfies all constraints. As originally defined in artificial intelligence, constraints enumerate the possible values a set of variables may take. Informally, a finite domain is a finite set of arbitrary elements.

5.2 PHYSIOLOGICAL VARIABLES FOR MEN AND WOMEN

All sports learners analyze physiological variable before undertaking an activity. The keyword (BHRhigh, VCmedium) is not sufficient for training activity; it is essential to identify the relevant physiological variable for the activity and should be satisfied through CSP.
The physiological variable should satisfy the constraints to achieve the goal. The sports learner may end up with undesired result in case an activity is taken when the physiological variable is high. The physiological variable is slightly difference between the range for men and women.

Tables 5.1 and 5.2 are used to measure the sports physiological variables range is slightly difference change in the training activity for women and men. The e-learner gets optimized result when the physiological variable values are satisfied before starting an activity. Detailed description of sports physiological variables for men and women as shown in appendix I.

**Table 5.1 Range of physiological variables for women**

<table>
<thead>
<tr>
<th>Name</th>
<th>PULSE RATE</th>
<th>RESPIRATORY RATE</th>
<th>BLOOD PRESSURE</th>
<th>BREATH HOLDING TIME</th>
<th>AVG MEAN</th>
<th>MEAN</th>
<th>RANGE</th>
<th>STANDARD DEVIATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rashi 1/2</td>
<td>69</td>
<td>13</td>
<td>80</td>
<td>138</td>
<td>25</td>
<td>61</td>
<td>70</td>
<td>100</td>
</tr>
<tr>
<td>A. Anirudha</td>
<td>70</td>
<td>16</td>
<td>79</td>
<td>126</td>
<td>25</td>
<td>62</td>
<td>70</td>
<td>103</td>
</tr>
<tr>
<td>D. Sridevi</td>
<td>65</td>
<td>14</td>
<td>79</td>
<td>138</td>
<td>22</td>
<td>62</td>
<td>65</td>
<td>104</td>
</tr>
<tr>
<td>M. Divya</td>
<td>68</td>
<td>15</td>
<td>80</td>
<td>120</td>
<td>31</td>
<td>63</td>
<td>68</td>
<td>105</td>
</tr>
<tr>
<td>K. Maheshwari</td>
<td>69</td>
<td>14</td>
<td>79</td>
<td>126</td>
<td>29</td>
<td>64</td>
<td>69</td>
<td>102</td>
</tr>
<tr>
<td>K. Kirti</td>
<td>69</td>
<td>15</td>
<td>79</td>
<td>118</td>
<td>31</td>
<td>63</td>
<td>68</td>
<td>105</td>
</tr>
<tr>
<td>N. Ponnamma</td>
<td>66</td>
<td>13</td>
<td>79</td>
<td>119</td>
<td>28</td>
<td>62</td>
<td>68</td>
<td>104</td>
</tr>
<tr>
<td>V. Veena</td>
<td>68</td>
<td>14</td>
<td>78</td>
<td>120</td>
<td>27</td>
<td>63</td>
<td>68</td>
<td>106</td>
</tr>
<tr>
<td>A. Valarmathy</td>
<td>70</td>
<td>16</td>
<td>79</td>
<td>119</td>
<td>26</td>
<td>61</td>
<td>70</td>
<td>101</td>
</tr>
</tbody>
</table>

**Table 5.2 Range of Physiological variables for men**

<table>
<thead>
<tr>
<th>Name</th>
<th>PULSE RATE</th>
<th>RESPIRATORY RATE</th>
<th>BLOOD PRESSURE</th>
<th>BREATH HOLDING TIME</th>
<th>AVG MEAN</th>
<th>MEAN</th>
<th>RANGE</th>
<th>STANDARD DEVIATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>T. Sathya</td>
<td>66</td>
<td>16</td>
<td>80</td>
<td>139</td>
<td>24</td>
<td>60</td>
<td>68</td>
<td>111</td>
</tr>
<tr>
<td>R. Srikan</td>
<td>66</td>
<td>16</td>
<td>81</td>
<td>128</td>
<td>24</td>
<td>60</td>
<td>68</td>
<td>110</td>
</tr>
<tr>
<td>M. Kamakshitha</td>
<td>66</td>
<td>18</td>
<td>78</td>
<td>119</td>
<td>31</td>
<td>63</td>
<td>68</td>
<td>106</td>
</tr>
<tr>
<td>R. Selvyn</td>
<td>68</td>
<td>14</td>
<td>83</td>
<td>127</td>
<td>36</td>
<td>66</td>
<td>68</td>
<td>111</td>
</tr>
<tr>
<td>R. Sree L.</td>
<td>65</td>
<td>13</td>
<td>85</td>
<td>129</td>
<td>40</td>
<td>65</td>
<td>66</td>
<td>116</td>
</tr>
<tr>
<td>C. Prachya</td>
<td>66</td>
<td>13</td>
<td>83</td>
<td>128</td>
<td>40</td>
<td>66</td>
<td>66</td>
<td>111</td>
</tr>
<tr>
<td>G. Tamil</td>
<td>66</td>
<td>15</td>
<td>84</td>
<td>127</td>
<td>40</td>
<td>66</td>
<td>66</td>
<td>111</td>
</tr>
<tr>
<td>B. Tarni</td>
<td>67</td>
<td>14</td>
<td>80</td>
<td>126</td>
<td>39</td>
<td>65</td>
<td>67</td>
<td>110</td>
</tr>
<tr>
<td>M. Varsha</td>
<td>68</td>
<td>15</td>
<td>78</td>
<td>119</td>
<td>38</td>
<td>64</td>
<td>69</td>
<td>106</td>
</tr>
<tr>
<td>M. K. Veerababu</td>
<td>69</td>
<td>14</td>
<td>81</td>
<td>126</td>
<td>38</td>
<td>63</td>
<td>69</td>
<td>110</td>
</tr>
<tr>
<td>D. Yamini 1/2</td>
<td>67</td>
<td>15</td>
<td>82</td>
<td>127</td>
<td>34</td>
<td>64</td>
<td>69</td>
<td>112</td>
</tr>
<tr>
<td>A. Vimal Sudhakar</td>
<td>66</td>
<td>13</td>
<td>80</td>
<td>126</td>
<td>40</td>
<td>65</td>
<td>66</td>
<td>111</td>
</tr>
</tbody>
</table>
5.3 CONSTRAINT SATISFACTION PROBLEM (CSP)

A solution to Constraint satisfaction problem is a complete assignment that satisfies all constraints. For example a set of variables $X_1$, $X_2$, ..., $X_n$ and a set of constraints $C_1, C_2, ..., C_m$. Each variable $X_i$ has a non empty domain $D_i$ of possible values. Each constraint $C_i$ involves some subset of the variables and specifies the allowable combination of values for that subset. A state of the problem is defined by an assignment of values to some or all variables $\{X_i = v_i, X_j = v_j, ...\}$. An assignment that does not violate any constraint is called a consistent or legal assignment.

In sports Domain the physiological variables are Speed(SP), Blood Pressure (BP(S,D)), Stamina(SD), Respiratory Rate(RR), Endurance(ED), Vital Capacity(VC), Heart Rate(I,E), Breath Holding Rate(BHR) and Resting Pulse Rate(RPR) and the domain values $D_i$ are Low(L), High(H), Normal(N), Medium(M), etc the constraint $C_i$ are the allowable constraints for Example in case of SP and BP(S), RR, SD the constraints are \{MHLN, MHNL, MLHN, MLNH, MNLH, HMLN, HMNL, HLMN, HLNM, HNML, HNLM, LMHN, LMNH, LHMN, LHNM, LNHM, NMHL, NMLH, NMLH, NHLM, NLMH, NLHM, NLHM\}.

Figure 5.1 shows the physiological variable problem represented as a constraint graph. The binary CSP has a constraint graph representation where nodes are variable and the edges denote the constraint between the variable pairs.

There are many possible solutions for CSP as shown in Figure 5.1. Such as \{SP=N, BP(S) =M, SD=H, RR=L, ED=N, VC=M, HR (I) =N, BHR=H, RPR=L and HR (E) =N\}.
Figure 5.1 Constraint graph using physiological variables

It is helpful to visualize a CSP as a constraint graph. The nodes correspond to the variables of the problem and the arcs correspond to the constraints.

5.4 TYPES OF CONSTRAINT

There are two types of constraints: Unary constraint and Binary constraint.

The simplest type is the Unary constraint, which restricts the value of a single variable. For example, it could be the case that Stamina actively dislikes the variable Low. Every unary constraint can be eliminated simply by preprocessing the domain of the corresponding variable, and removing any value that violates the constraint.

A binary constraint relates two variables. For example, SD not= RR is a binary constraint. A binary CSP has only binary constraints and is represented as a constraint graph.
5.5 BACKTRACKING SEARCH

Once CSP is formulated the search algorithm is used for solving CSP. Suppose if Breadth first search (BFS) is applied for a CSP the branching factor at top level is nd since any of d values can be assigned to n variables, in the next level the branching factor is (n-1)d and so on for the n levels. This generates a tree with n! d^n leaves even though only d^n possible complete assignments. This ignores the crucial property of CSP which is commutative which means there is no impact on the outcome based on the order of application of given set of actions.

All CSP search algorithms generate successors by considering possible assignments for only a single variable at each node in the same search tree. For example at root node a search trees for a physiological variable the choice between High, Low, Medium, and Normal.

Backtracking is used with depth first searching (DFS) in which value for one variable at a time is chosen at a time and backtracks when a variable does not have any legal value left for to assign.

In Figure 5.2 shows the search tree generated backtracking

Algorithm 1

Function Backtracking Search

**FBS** (Stage S, Variables W, Constraint C)  {returns false or true}

Domain k, X= {N, M, L,H}, W= {SP, BP(S), SD, RR}

if all **ASSIGNED** in W

W ← x_i  \[x_i \in X, i=1…n]\)

else

y ← UA(S,W,C)  \{ Unassigned variable \)
for each $x_i$ in STAGE $(y,W,C)$ do  

STAGE function contains order of domain $k$  

if $x_i$ is CONSISTENT with $W$ according to $C$ then  

W is the assignment of Variable SP, BP(S), SD, RR}  

ADD{$y=x_i$} $\leftarrow W$;  

R $\leftarrow$ FBS($S,W,C$)  

{R stands for result}  

if $R \neq false$ then return $R$  

REMOVE $(y=x_i)$ $\rightarrow W$  

return false

It has different levels of input stages s(levels). Domain $k$ has set of set of variables $W$ = {SP, BP(S), SD, RR} The variables are assigned with values one at time in DFS mode. When an assignment is not legal the DFS is back tracked. In this case SP is assigned at $N,M,L,H$ in stage 1 and in stage 2 BP(s) is assigned values MLH, NLH, NMH & NML and other combinations of assignment are illegal hence back tracked.
5.6 FORWARD CHECKING

Forward checking keeps track of remaining legal values for unassigned variables. It terminates the search when any variable has no legal values. Table 5.3 represents the forward checking used in the model.

**Table 5.3 Forward checking for different variables**

<table>
<thead>
<tr>
<th></th>
<th>SP NMHL</th>
<th>BP(S) NMHL</th>
<th>ED NMHL</th>
<th>HR(E) NMHL</th>
<th>RPR NMHL</th>
<th>BHR NMHL</th>
<th>HR(I) NMHL</th>
<th>SD NMHL</th>
<th>RR NMHL</th>
<th>VC NMHL</th>
</tr>
</thead>
<tbody>
<tr>
<td>After SP Normal(N)</td>
<td>N</td>
<td>MHL</td>
<td>NMHL</td>
<td>NMHL</td>
<td>NMHL</td>
<td>NMHL</td>
<td>NMHL</td>
<td>MHL</td>
<td>MHL</td>
<td>NMHL</td>
</tr>
<tr>
<td>After HR(I) Medium(M)</td>
<td>N</td>
<td>NMHL</td>
<td>NMHL</td>
<td>NMHL</td>
<td>N HL</td>
<td>N HL</td>
<td>M</td>
<td>N HL</td>
<td>N HL</td>
<td>N HL</td>
</tr>
<tr>
<td>After ED High(H)</td>
<td>N</td>
<td>NM L</td>
<td>H</td>
<td>NM L</td>
<td>NMHL</td>
<td>NMHL</td>
<td>NMHL</td>
<td>NMHL</td>
<td>NM L</td>
<td>NM L</td>
</tr>
<tr>
<td>After RPR Low(L)</td>
<td>N</td>
<td>NMHL</td>
<td>NMH</td>
<td>NMH</td>
<td>L</td>
<td>NMH</td>
<td>NMH</td>
<td>NMHL</td>
<td>NMHL</td>
<td>NMH</td>
</tr>
</tbody>
</table>

First assign {SP=normal} means that the value of SP is normal and the variable connected to the constraint doesn’t hold the value normal. i.e. BP(S), RR, SD can no longer be normal.

Second assign {HR(I)=medium} means that the value of HR(I) is medium and the variables connected to the constraints doesn’t hold the value of SP. SD,RR,VC,BHR,RPR can no longer be medium. So the heuristic would automatically select RR and SD.

Third assign {ED=high} means that the value of ED is high and the variables connected to the constraints doesn’t hold the value of SP. BP(S), RR, VC, HR(E), RPR can no longer be high. So the heuristic would automatically select BP(S), RR.
Fourth assign \(\{RPR=\text{low}\}\) means that the value of RPR is low and the variables connected to the constraints doesn’t hold the value of SP. BHR, HR(I), VC, HR(E), ED can no longer be low. So the heuristic would automatically select HR(E), ED, VC. Forward Checking has detected that partial assignment is inconsistent with the constraints and backtracking can occur.

5.7 CONSTRAINT PROPAGATION

Propagating implications of a constraint on one variable to other variables is constraint propagation the ARC consistency provides a fast propagation

Arc Consistency

Arc consistency operates on constraints, variables, and the variables domains. A variable can take any of the several discrete values; the set of values for a particular variable is known as its domain. A constraint is a relation that limits or constrains the values of a variable. The constraint involves the values of other variables. Arc consistency refers to directed arc in constraint graph.

Function Arc Consistency (FAC) is given below.

Function Arc Consistency

\[
\text{FAC} \ (\text{Domain C, value of variables } \{W= w_1, w_2, w_3, \ldots, w_n, i=1\ldots n}\})
\]

Output: CSP

Local variables: Queue S, Edges in C

While \(S \neq \text{NULL}\) do
\[(w_i, w_j) \leftarrow \text{DEL-FIRST}(S)\]

\textbf{if} \ \text{INCON}(w_i, w_j) \ \textbf{then} \\
\textbf{for} each \ w_h \ \text{in ADDJ}[w_i] \ \textbf{do} \\
\ \ \ \text{ADDQ} \ (w_h, w_j) \leftarrow S \\
\textbf{Function INCON}(w_i, w_j) \quad \{\text{returns true if value is removed}\} \\
\ \ \ \text{DELETED} \leftarrow \text{False} \\
\textbf{for} each \ w_i \ \text{in} \ W \ \text{do} \\
\textbf{if} \ y \neq w_j \ \textbf{then} \\
\ (w_i, y) \ \text{satisfies constraint} \ C(w_i, w_j) \\
\ \ \ \text{DEL} \ (w_i, W) \\
\ \ \ \text{DELETED} \leftarrow \text{true} \\
\ \ \ \text{Return DELETED} \\

In the Arc consistency function the consistency checking process is repeated until all the inconsistencies are removed. Whenever a value is deleted from variables domain to remove arc inconsistency possibly a new arc inconsistency may arise due to secondary effect, the algorithm uses a queue to keep track of arcs that need to be checked for inconsistency for example the ARC from RR to VC. Given the current domains of RR and VC the arc is consistent if, for every value of \(x\) of RR, there is some value \(y\) of VC that is consistent with \(x\). For \(RR=\text{Low}\) there is a consistent assignment of \(VC=\text{Medium}\) therefore an arc from RR to VC is consistent on the other hand the reverse from VC to RR is not consistent.
5.8 CONSTRAINT STRUCTURE FOR SPORTS PERSON PHYSIOLOGICAL VARIABLES

Consider the constraint graph shown in Figure 5.3. The value of BHR and HR (I) is the same. Hence, the constraints are not satisfied. In order to satisfy the constraint, the nodes BHR, HR (I) are deleted. After deletion the node can solve the whole problem by using this approach. The value chosen for BHR could be a wrong one and it needs to try each of them. The values are deleted as shown in Figure 5.3.

![Figure 5.3 The constraint graph after the removal of HR (I) and BHR](image-url)
The backtracking and consistency propagation employed here improves the performance and in turn efficient and precise result. The constraint structured, is framed after the thorough study of the physiological variables considered.